UNITED STATES PATENT AND TRADEMARK OFFICE

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/782,850	02/14/2001	Steven Mark Gebert	BLD920000048US1	9299

			EXAMINER
46919	7590	08/23/2006	CAMPBELL, JOSHUA D

KONRAD RAYNES & VICTOR, LLP.
ATTN: IBM36
315 SOUTH BEVERLY DRIVE, SUITE 210
BEVERLY HILLS, CA 90212

ART UNIT	PAPER NUMBER
2178	

DATE MAILED: 08/23/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

UNITED STATES PATENT AND TRADEMARK OFFICE

MAILED

AUG 2 3 2006

Technology Center 2100

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Application Number: 09/782,850
Filing Date: February 14, 2001
Appellant(s): GEBERT ET AL.

David Victor
<u>For Appellant</u>

EXAMINER'S ANSWER

This is in response to the appeal brief filed May 3, 2006 appealing from the Office action

mailed November 28, 2005.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

The appellant's statement of the status of amendments after final rejection contained in the brief is incorrect.

The amendment after final rejection filed on May 2, 2006 has been entered.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

Adler et al. "Extensible Stylesheet Language (XSL) Version 1.0" W3C, Working

Draft (October 18, 2000), pp. 1-29

Sall, "FOP: Formatting Object to PDF Translator (James Tauber)" (May 24,

1999), pp. 1-3

| 5,323,312 | Saito et al. | 6-1994 |
| 6,606,165 | Barry et al. | 1-1999 |

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claims 1, 2, 4, 8-11, 14-16, 18, 22-25, 28-30, 32, 36-39, and 42-48 are rejected under

35 U.S.C. 103(a) as being unpatentable over Adler et al. (hereinafter Adler, Extensible

Stylesheet Language (XSL) Version 1.0, published on October 18, 2000) in view of

Saito et al. (hereinafter Saito, US Patent Number 5,323,312, issued on June 21, 1994).

Regarding independent claim 1, Adler discloses a method in which a source

document including source content is received in XML (pages 17-18, section 1.1

Processing a Stylesheet). Then, a layout data structure (XSL stylesheet) which

provides formatting properties and is separate from the source document and does not

contain source content is received (pages 17-18, section 1.1 Processing a Stylesheet).

The two documents are processed together and to determine formatting, including page

divisions, of the source content (pages 20-21, Section 1.1.2 Formatting and pages 25-

27, Section 1.2.1 Paging and Scrolling and Section 1.2.3 An Extended Page Layout

Model). Adler also discloses a method in which multiple page objects are generated by

filling the XML content into "containers", each of the objects containing the content and

the information required to format the content, at which point the "containers" are

rasterized into page instances which are capable of being generated by an output

device (pages 20-21, Section 1.1.2 Formatting and pages 25-27, Section 1.2.1 Paging

and Scrolling and Section 1.2.3 An Extended Page Layout Model). Adler discloses that

the page objects include the content that is to be placed on the pages, Adler does not

explicitly state that the content consists of multiple content elements. However, Saito

discloses that it was well known in the art that a structured document could consist of

two parts a layout structure and a logical structure (source content), and when filling the

page objects defined by the layout structure more than one content object from the

logical structure could be used, thus allowing more than one content object to exist

within each page object (column 1, lines 31-57 of Saito). It would have been obvious to

one of ordinary skill in the art at the time the invention was made to have used the

method of Adler with the well-known material disclosed by Saito because it would have

allowed the space in visible pages to be used completely by filling page objects with

more than content object where it was warranted.

Regarding dependent claims 2 and 4, Adler also discloses a method in which

the source document (XML) and the result document (XSL-FO) may be different

formats, and the result document is formatted based on the layout data structure (XSL) (pages 17-18, section 1.1 Processing a Stylesheet and pages 20-21, Section 1.1.2 Formatting). Adler also discloses a method in which multiple page objects are generated by filling the XML content into "containers", each of the objects containing the content and the information required to format the content, at which point the "containers" are rasterized into page instances which are capable of being generated by an output device (pages 20-21, Section 1.1.2 Formatting and pages 25-27, Section 1.2.1 Paging and Scrolling and Section 1.2.3 An Extended Page Layout Model).

Regarding dependent claims 8-10, Adler discloses a method in which page divisions may be presented in XSL-FO, which is a device independent language (pages 20-21, Section 1.1.2 Formatting and pages 25-27, Section 1.2.1 Paging and Scrolling and Section 1.2.3 An Extended Page Layout Model).

Regarding dependent claim 11, Adler discloses a method in which a page description language is used (pages 17-18, section 1.1 Processing a Stylesheet).

Regarding dependent claim 14, Adler discloses a method in which the source document does not indicate page divisions (pages 17-18, section 1.1 Processing a Stylesheet).

Regarding dependent claims 43-44, Adler does not explicitly disclose that a page object is filled with content objects until there is no more space, then the next page element is filled with the sequential content objects, or that page sequence elements exist in which the content is accessed in sequence and added to the page objects accordingly. However, Saito discloses that it was well known in the art that a structured

document could consist of two parts a layout structure and a logical structure (source

content in the logical page viewing sequence), and when filling the page objects defined

by the layout structure more than one content object from the logical structure could be

used per page until a page is full at which point the next page object is filled with

content and so forth (column 1, lines 31-57 of Saito). It would have been obvious to one

of ordinary skill in the art at the time the invention was made to have used the method of

Adler with the well-known material disclosed by Saito because it would have allowed the

space in visible pages to be used completely by filling page objects before moving on to

filling the next sequential page object.

Regarding independent claim 15 and dependent claims 16, 18, 22-25, 28, 45,

and 46, the claims incorporate substantially similar subject matter as claims 1, 2, 4, 8-

11, 14, 43, and 44. Thus, the claims are rejected along the same rationale as claims 1,

2, 4, 8-11, 14, 43, and 44.

Regarding independent claim 29 and dependent claims 30, 32, 36-39, 42, 47,

and 48, the claims incorporate substantially similar subject matter as claims 1, 2, 4, 8-

11, 14, 43, and 44. Thus, the claims are rejected along the same rationale as claims 1,

2, 4, 8-11, 14, 43, and 44.

Claims 5, 7, 19, 21, 33, and 35 are rejected under 35 U.S.C. 103(a) as being

unpatentable over Adler et al. (hereinafter Adler, Extensible Stylesheet Language (XSL)

Version 1.0, published on October 18, 2000) in view of Saito et al. (hereinafter Saito, US

Patent Number 5,323,312, issued on June 21, 1994) as applied to claims 2, 16, and 30

above, and further in view of Barry et al. (hereinafter Barry, US Patent Number

6,606,165, filed on January 8, 1999).

Regarding dependent claim 5, 7, 19, 21, 33, and 35, Adler does not disclose

page objects which are in a third presentation language which is a page description

language. However, Barry discloses a method in which a document is split into multiple

page objects that contain the source content and formatting for one page in a different

page description language (image bit-map) (column 1, line 24- column 3, line 11 of

Barry). It would have been obvious to one of ordinary skill in the art at the time the

invention was made to have combined the method of Adler with the method of Barry

because it would have simplified the use of an output device to render a multi-paged

document.

Claims 6, 20, and 34 are rejected under 35 U.S.C. 103(a) as being unpatentable over

Adler et al. (hereinafter Adler, Extensible Stylesheet Language (XSL) Version 1.0,

published on October 18, 2000) in view of Saito et al. (hereinafter Saito, US Patent

Number 5,323,312, issued on June 21, 1994) further in view of Barry et al. (hereinafter

Barry, US Patent Number 6,606,165, filed on January 8, 1999) as applied to claims 5,

19, and 33 above, and further in view of Sall (as found in the IDS - FOP: Formatting

Object to PDF Translator (James Tauber, published in 1999).

Regarding dependent claims 6, 20, and 34, Adler also discloses a method in

which the source document (XML) and the result document (XSL-FO) may be different

formats, and the result document is formatted based on the layout data structure (XSL)

(pages 17-18, section 1.1 Processing a Stylesheet and pages 20-21, Section 1.1.2

Formatting). Adler also discloses a method in which multiple page objects are

generated by filling the XML content into "containers", each of the objects containing the

content and the information required to format the content, at which point the

"containers" are rasterized into page instances which are capable of being generated by

an output device (pages 20-21, Section 1.1.2 Formatting and pages 25-27, Section

1.2.1 Paging and Scrolling and Section 1.2.3 An Extended Page Layout Model). Adler

does not disclose a method in which the language of the page objects is MO:DCA, a

common presentation imaging language. However, Sall discloses a method in which an

XML is converted to XSL-FO based on an XSL stylesheet, then based on XSL-FO

convert the document to a PDF (pages 1-2 of Sall), which as defined in 1998 by

McCalpin (page 3 of "Traditional Electronic Printing on the Internet") as being an

common analogous presentation language to MO:DCA. It would have been obvious to

one of ordinary skill in the art at the time the invention was made to have combined

Adler with methods taught by Sall because this method was noted to be a potential

replacement for typical desktop published due to formatting advantages.

(10) Response to Argument

Regarding the appellants arguments on pages 5-10, which are in reference to the

claim limitation "...generating multiple page objects, wherein each page object includes

the source content in the presentation language used in the source document and the

determined formatting properties for one page, wherein at least one page object has

multiple content elements," the examiner maintains that the invention as claimed is properly rejected. Adler discloses that a result tree is generated by processing the XSL document (layout data structure) and the source tree (source content) (page 18, Final Paragraph and Figure of Adler). Adler explicitly discloses that the result tree consists of objects in the "formatting object" namespace (page 18, Final Paragraph). Adler teaches that, "Formatting semantics are expressed in terms of a catalog of classes of formatting objects. The nodes of the result tree are formatting objects. The classes of formatting objects denote typographic abstractions such as _page_, paragraph, table, and so forth," (page 18, first paragraph, lines 1-4 of Adler). Adler also teaches that the formatting objects are represented as XML elements with the properties and attributes of the XML value pairs and the content of the original XML element (i.e. original source content) (page 18, Final Paragraph of Adler). This process is more clearly explained using the figure on Page 19, which shows the source tree (original XML content) is transformed using XSL stylesheet, which represents the formatting information for the XML content. This process creates the result tree which has formatting objects for nodes which included the original XML content and formatting information necessary to properly display the content. In other words, the result of processing the source content and the layout data structure is a result tree. This result tree consists of formatting objects which correspond to typographic abstractions such as pages, also known as "page objects," if the document being processed consisted of more than one page of data the result tree would be required to have multiple page objects. These objects are stand-alone abstractions which contain both the formatting and the content necessary for each

typographic abstraction, once again in this case that would be a page. The latter part of

the limitation, "...wherein at least one page object has multiple content elements, and

wherein the content elements include content to place on the pages," is not explicitly

taught by Adler, which is why the 35 U.S.C 103(a) rejection was made in view of Saito.

Saito discloses that it was well known in the art that a structured document could

consist of two parts a layout structure and a logical structure (source content), and when

filling the page objects defined by the layout structure more than one content object

from the logical structure could be used, thus allowing more than one content object to

exist within each page object (column 1, lines 31-57 of Saito). In other words Saito

teaches that when pages objects are made, more than one content object can be

placed on the page. Any document that has text and an image or table would fall into

the category of having multiple content elements, which is why these teachings of Saito

are believed to be well known to one of ordinary skill in the art at the time of invention by

the appellant. However, motivation to combine was still provided, it would have allowed

the space in visible pages to be used completely by filling page objects with more than

content object where it was warranted. In other words, if a content object did not fill a

page and you only put one object per page you would end up with wasted white space

on every page of a document, but as shown by the teachings of Saito by allowing more

than one content object to appear on each page, the wasted space is greatly reduced.

Regarding the appellants arguments on pages 11, paragraph 1 that are in

reference to rasterizing the objects into render-able information, the examiner maintains

that the invention as claimed is properly rejected. The Figure on page 18 of Adler

clearly shows the result tree is transmitted to a printer if that is the selected output. The definition of rasterize as found using the Google search engine's definition finder states, "All computer files are rasterized when they're printed," (search page attached as an appendix to the examiner answer). Thus, being that the result tree is printed it is clearly rasterized.

Regarding the appellants arguments on pages 11, paragraph 2-page 12 that are in reference to the claim limitation, "...wherein at least one page object has multiple content elements, and wherein the content elements include content to place on the pages," the examiner maintains that the invention as claimed is properly rejected. As previously stated Adler does not explicitly teach the limitation, which is why the 35 U.S.C 103(a) rejection was made in view of Saito. Saito discloses that it was well known in the art that a structured document could consist of two parts a layout structure and a logical structure (source content), and when filling the page objects defined by the layout structure more than one content object from the logical structure could be used, thus allowing more than one content object to exist within each page object (column 1, lines 31-57 of Saito). In other words Saito teaches that when pages objects are made, more than one content object can be placed on the page. Any document that has text and an image or table would fall into the category of having multiple content elements, which is why these teachings of Saito are believed to be well known to one of ordinary skill in the art at the time of invention by the appellant. However, motivation to combine was still provided, it would have allowed the space in visible pages to be used completely by filling page objects with more than content object where it was warranted.

In other words, if a content object did not fill a page and you only put one object per page you would end up with wasted white space on every page of a document, but as shown by the teachings of Saito by allowing more than one content object to appear on each page, the wasted space is greatly reduced.

Regarding the appellants arguments on pages 13, paragraph 2-page 14, paragraph 2, these arguments are admittedly the same as the first set of presented arguments, the rationale for maintaining the rejection can be found above.

Regarding the appellants arguments on page 14, paragraphs 3 and 4, that are in reference to the claim limitation, "...wherein the multiple page objects are generated from the result document," the examiner maintains that the invention as claimed is properly rejected. Adler discloses an ability of XSL known as the Extended Page Layout Model. In this model, once a result tree exists "simple-page-masters" may be used to which content is used to fill pages and how the styled content is to be placed regionally on the page (page 27, section 1.2.3 of Adler). These "simple-page-masters" are understood to be advanced page objects that have the ability to take a result tree and break it into page objects, whether it had predefined page abstractions or not. Thus, clearly showing that page objects could indeed be generated from the result tree.

Regarding the appellants arguments on page 15, that are in reference to page objects being generated from the result document including source content in a first language and formatting properties in a second language, the examiner maintains that the invention as claimed is properly rejected. Appellant's arguments fail to comply with 37 CFR 1.111(b) because they amount to a general allegation that the claims define a

patentable invention without specifically pointing out how the language of the claims

patentably distinguishes them from the references. As previously stated, Adler

discloses an ability of XSL known as the Extended Page Layout Model. In this model,

once a result tree exists "simple-page-masters" may be used to which content is used to

fill pages and how the styled content (original XML content) is to be placed regionally on

the page using XSL (the second presentation language) (page 27, section 1.2.3 of

Adler). Thus, clearly showing page objects being generated from the result document

including source content in a first language and formatting properties in a second

language.

Regarding the appellants arguments on page 16, that are in reference to

including content elements in page sequence elements to determine how to add content

elements to pages, the examiner maintains that the invention as claimed is properly

rejected. Appellant's arguments fail to comply with 37 CFR 1.111(b) because they

amount to a general allegation that the claims define a patentable invention without

specifically pointing out how the language of the claims patentably distinguishes them

from the references. Saito discloses that it was well known in the art that a structured

document could consist of two parts a layout structure and a logical structure (source

content in the logical page viewing sequence), and when filling the page objects defined

by the layout structure more than one content object from the logical structure could be

used per page until a page is full at which point the next page object is filled with

content and so forth (column 1, lines 31-57 of Saito). These teachings combined with

the Extended Page Layout Model teachings in Adler (page 27, section 1.2.3 of Adler)

render the claimed limitations obvious. Motivation to combine was provided, it would have allowed the space in visible pages to be used completely by filling page objects with more than content object where it was warranted. In other words, if a content object did not fill a page and you only put one object per page you would end up with wasted white space on every page of a document, but as shown by the teachings of Saito by allowing more than one content object to appear on each page, the wasted space is greatly reduced.

Regarding the appellants arguments on page 16, that are in reference to claims 5, 19, and 33, the examiner maintains that the invention as claimed is properly rejected. Appellant's arguments fail to comply with 37 CFR 1.111(b) because they amount to a general allegation that the claims define a patentable invention without specifically pointing out how the language of the claims patentably distinguishes them from the references. Barry discloses a method in which a document is split into multiple page objects that contain the source content and formatting for one page in a different page description language (image bit-map) (column 1, line 24- column 3, line 11 of Barry). The use of image bit-map in combination with the teachings of Adler as shown above would constitute a third presentation language. The advantage of using a image bit-map as the final presentation language for the page objects is that it would have simplified the use of an output device to render a multi-paged document. It would have been obvious to one of ordinary skill in the art at the time the invention was made to have made this combination of references.

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the

Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

Joshua Campbell

STEPHEN HONG

SUPERVISORY PATENT EXAMINER

Conferees:

Stephen Hong, Supervisory Patent Examiner for Group Art Unit 2178

Heather Herndon, Supervisory Patent Examiner for Group Art Unit 2176

Heather R. Herndon
Supervisory Patent Examiner
Technology Center 2100

Google

Web   **Images**   Video^{New!}   News   Maps   **more »**

define: rasterize   Search   Advanced Search
                                              Preferences

Web

Definitions of **rasterize** on the Web:

- The conversion of an object-oriented image into a bitmapped image. When an Adobe Illustrator graphic is placed or opened in PhotoShop, it is rasterized. All computer files are rasterized when they're printed.
  www.zacsdesign.com/edu/basic_multimedia_glossary.htm

- The conversion of a vector image into a bit-mapped image.
  www.fotoserve.com/glossary/r.html

- To render vector data into a bitrmap image.
  www.wizzywigg.com/lev2_glossary5.htm

- convert (an image) into pixels
  wordnet.princeton.edu/perl/webwn

- A raster graphics image, digital image, or bitmap, is a data file or structure representing a generally rectangular grid of pixels, or points of color, on a computer monitor, paper, or other display device. The color of each pixel is individually defined; images in the RGB color space, for instance, often consist of colored pixels defined by three bytes—one byte each for red, green and blue. ...
  en.wikipedia.org/wiki/Rasterize

define: rasterize   Search

Language Tools | Search Tips | Dissatisfied? Help us improve

Google Home - Advertising Programs - Business Solutions - About Google

©2006 Google

Appendix A